March 28, 2005

Report of the Fermilab Computing Division 64-bit Linux Evaluation Taskforce

H. Wenzel, A. Singh, S. Timm, K. Shepelak, B. Forster,D. Fagan,J. Kaiser,D. Holmgren,P. Lutz, R. Van Conant, K. Ruthmansdorfer, F. de Jongh, R. Kessler.

Fermi National Accelerator Laboratory, Batavia, USA

University of Chicago, Chicago, USA

Abstract

The Fermilab Computing Division 64-Bit Linux Evaluation Taskforce has evaluated the AMD64 (x86_64) and Intel EM64T architectures for use at Fermilab. We tested the machines for reliability, performance, and energy consumption, using real applications from CMS, Run II experiments, SDSS, and LQCD. This report summarizes the new features of the AMD64 and EM64T architectures. It then presents results of the reliability, performance, and power consumption tests. It is shown that for the considered applications, the available 64-bit commodity computers from AMD and Intel are a viable alternative to comparable 32-bit systems, and that the x86_64 release of Scientific Linux is a viable operating system to operate on. Significant gains in performance might be achieved by porting software to new processor platforms in 64-bit mode. In addition, we find the reliability of the Opteron-based systems to be adequate for Fermilab applications, and the power consumption to be significantly less than their Intel counterparts.

Preliminary version

1 Introduction

A nice introduction to the AMD64 and EM64T as well as various benchmark results can be found in [1]. There are now three 64-bit implementations in the Intel compatible processor marketplace:

- Intel IA64, as implemented on the Itanium 2 processor. Since it doesn't provide a compatibility mode with the current IA32 platform we will not be covering this architecture in this note.
- Intel EM64T, as implemented on the Xeon DP Nocona and future Xeon MP processors.
- AMD AMD64, as implemented on the Opteron processor.

Given that Intel and AMD have already shifted production so that the majority of their output are 64-bit processors, even on the desktop, 64-bit hardware technology is inevitable. There are already 216 units of 64-bit Intel Xeon EM64T hardware on site. At the moment they are running in 32-bit OS mode. They have passed a full farms burn-in and are in production now for Minos, D0 and CDF Reconstruction Farms, the D0 CAB, and the CDF CAF. The purpose of this evaluation is two-fold. We are evaluating the stability of the x86_64 architecture of Linux and the stability of the AMD Opteron hardware platform.

In this note we present benchmark results obtained with EM64T and AMD64. As an operating system we used Scientific Linux 3.03 which is available in 32 Bit and 64 Bit.

There are three distinct operation modes of interest available in AMD64 and EM64T:

- **32-bit legacy mode (32Bit/32Bit)**

  In this mode, both AMD64 and EM64T processors will act just like any other IA32 compatible processor. You can install your 32-bit OS on such a system and run 32-bit applications, however, you will not be able to make use of the new features such as the flat memory addressing above 4 GB or the additional General Purpose Registers (GPRs). 32-bit applications will run just as fast as they would on any current 32-bit processor.

  Most of the time IA32 applications will run even faster since there are numerous other improvements that boost performance regardless of the maximum address size. For applications that share large amounts of data, there might be performance impacts related to the NUMA-like architecture of multi-processor Opteron configurations, since remote memory access might slow your application down. If configured correctly, the NUMA can work to the advantage of the application as well.

- **Compatibility mode (64Bit/32Bit)**

  The second mode supported by the AMD64 and EM64T is compatibility mode, which is an intermediate mode of the full 64-bit mode described below. In order to run in compatibility mode, you will need to install a 64-bit operating system and 64-bit drivers. If a 64-bit OS and drivers are installed, both Opteron and Xeon processors will be enabled to support a 64-bit operating system with both 32-bit applications or 64-bit applications.

  Compatibility mode gives you the ability to run a 64-bit operating system while still being able to run unmodified 32-bit applications. Each 32-bit application will still be limited to a maximum of 4 GB of physical memory. However, the 4 GB limit is now imposed on a per-process level, not at a system-wide level. This means that every 32-bit process on this system gets its very own 4 GB of physical memory space (assuming sufficient physical memory is installed). This is already a huge improvement compared to IA32, where the operating system kernel and the application have to share 4 GB of physical memory.

- **Full 64-bit mode (64Bit/64Bit)**

  The final mode is the full 64-bit mode. AMD refers to this as long mode and Intel refer to it as IA-32e mode. This mode is when a 64-bit operating system and 64-bit application are used. In the full 64-bit operating mode, an application can have a virtual address space of up to 40-bits (that equates to 1 TB of addressable memory).

  Applications that run in full 64-bit mode will get access to the full physical memory range, and will also get access to the new GPRs as well as to the expanded GPRs. However it is important to understand that this mode of operation requires not only a 64-bit operating system (and of course 64-bit drivers), but also requires a 64-bit application that has been recompiled to take full advantage of the various enhancements of the 64-bit addressing architecture.

All available AMD64 operating modes are summarized in Figure 1. Figure 2 shows the additional registers available on the AMD64 processor architecture.

Operating Mode		OS Required	Application Recompile Required	Defaults		Register Extensions	Typical GPR Width
				Address Size (bits)	Operand Size (bits)		
Long Mode	64-bit Mode	New 64-bit OS	yes	64	32	yes	64
	Compatibility Mode		no	32		no	32
				16	16		16
Legacy Mode	Protected Mode	Legacy 32-bit OS	no	32	32	no	32
				16	16		
	Virtual-8086 Mode			16	16		16
	Real Mode	Legacy 16-bit OS					

Figure 1: *AMD64 operating modes (from [6])*

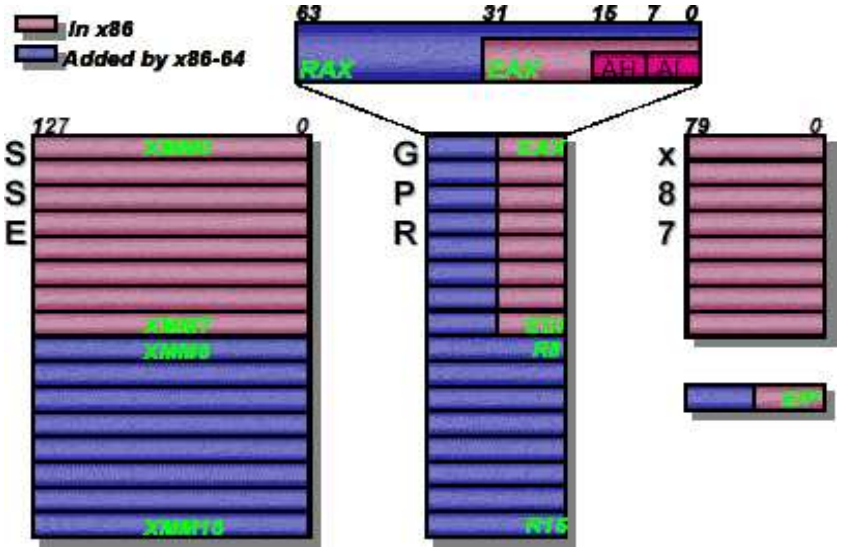

Figure 2: *Additional registers added in AMD64 [6])*

2 Vendor Selection and Machine Configuration

Vendors were invited to participate in this evaluation based on past experience with Fermilab. We requested twelve vendors to submit a machine to this evaluation. These were our current top six vendors for Intel hardware, Koi, ASA, Rackable, CSI, Penguin, and Dell, and those other vendors which were part of our eighteen technically qualified vendors from the previous evaluation. These include HP, IBM, Aspen, Promicro, and APPRO. Since Sun has already sold some Opteron based servers to Fermilab and Fermilab has a long-standing relationship with Sun we invited Sun to participate as well.

Vendors were each asked to provide a machine with 2 GB of SDRAM, and two disks (80 GB and 250 GB). Each vendor was asked to provide a specific speed of processor so that we were ensured a variety of processor speeds. We ended up having Opteron 244,246, 248, and 250 models available, corresponding from 1.8 to 2.4 GHz processor speed. AMD makes a low-voltage Opteron chip known as the 246HE, which has 55W thermal design power as opposed to its full-power counterpart. Unfortunately the HE chip is not available in all speeds.

Eight vendors, (Penguin, Rackable, ASA, CSI, IBM, Koi, Sun, and HP) successfully delivered an Opteron unit.

Dell, which does not make Opteron units, was asked to provide the fastest available 64-bit Intel Xeon unit, a 3.6GHz based Dell Poweredge SC1425, for comparison purposes. We also had a 3.4GHz Xeon unit available for comparison. The other vendors did not respond to our request for quote.

We saw five major types of motherboard. These are the Tyan S2882, the Accelertech (formerly Rioworks) HDAMA board, the Newisys board and server combination, the IBM board which is a proprietary board, and the HP solution. This covered all the architectures available in December when we made the orders. Since then a new board has been released by Iwill, of interest due to its on-board PCI express and infiniband connectors, and another has come from Supermicro which claims to support IPMI 2.0.

All the boards in this sample used some variant of the Broadcomm gigabit ethernet controller, and came with two gigabit ethernet interfaces. The Tyan board comes with a third 100-mbit Intel interface as well. Sun, IBM, Dell and HP also have a management net port as well for out-of-band management, all of which claim to be IPMI compatible. The Koi unit has an add-in IPMI card that supports IPMI 1.5.

We had different vendors available giving similar results when equipped with the same processor. Therefore the results here are listed as a function of processor type. In particular we had four different Opteron 248 machines, the results from only one of them are shown in most of the results of this report, but in most cases the results were quite similar between the four machines. All machines tested here are dual processor machines. The 246HE denotes the AMD Opteron 246HE processor which is a low power version of the Opteron which can be found in various Opteron based Blade servers. We were interested in how this processor performs compared to the "normal" Opteron. The IA32 machines represent our current worker nodes and are listed here for comparison and reference.

Processor	Speed	Architecture
Opteron 244	1.8 GHz	AMD-64
Opteron 246	2.0 GHz	AMD-64
Opteron 246HE	2.0 GHz	AMD-64
Opteron 248	2.2 GHz	AMD-64
Opteron 250	2.4 GHz	AMD-64
Xeon	3.4 GHz	EM64T
Xeon	3.6 GHz	EM64T
Xeon	2.4 GHz	IA32
Xeon	2.8 GHz	IA32
Xeon	3.06 GHz	IA32
Pentium III	1.0 GHz	IA32

Table 1: *Different processors tested.*

3 Operating System Installation

We installed three different operating system partitions on these nodes. We used NPACI-Rocks 3.3.0 to install all of the systems via the network, installing an 64-bit architecture which is compatible with 64-bit Scientific Linux Fermi. Most of the reliability testing was done with this operating system and kernel booted. We also installed 32-bit (i386) Scientific Linux Fermi 3.0.3 on a second partition. A considerable amount of benchmark running as well as some reliability testing was done in this mode. We also tested operations with the XFS filesystem kernel and found them to work. Finally we installed what was at the time Scientific Linux Fermi 3.0.rolling on a third partition, and then installed the 2.6.9 kernel and its associated utilities on top of that. This was needed for the LQCD benchmark testing and to make use of the non-uniform memory access (NUMA) features which are available in that kernel. All the systems are capable of network booting and installation, and controlling that network boot across a Cyclades PM-10 serial console. We were able to identify 32-bit compatibility libraries for 64-bit mode for most of the applications in our benchmark set, even those that had not been tested by their authors in this mode before.

4 Reliability Testing

All nine systems were installed in a rack in Feynman Computing Center and put through the Fermilab acceptance test for 30 days. This includes two instances of seti@home, bonnie every ten minutes, and filling the whole disk once per day. We had one node which experienced two failures of the system disk hda, the second one of which they were not able to fix before the end of the 30-day burnin period. In Fermilab burnin language this node would be referred to as a lemon and this vendor has been disqualified. If the performance of the other seven Opteron nodes is considered, there are 210 possible days of uptime, and there were five incidents that caused a reboot of a node over that period of time. This corresponds to 97.6% uptime. One of these five incidents was due to a known problem with a disk. On the other four, we have no information at all on what caused them and have not been able to reproduce them since. The systems in question have been running the burnin consistently since Feb. 24 and we have not seen the hang again. The vendors in question are prepared to work with us further if we are able to reproduce this error. As the taskforce we believe the level of reliability of the Opteron-based systems we have observed is sufficient for mass deployment at Fermilab.

5 Power Consumption

Current was measured using a clamp-on meter, and checked with a kill-a-watt inline meter. At full load the machines were running in 64-bit mode and two copies of seti@home. We measured both with and without bonnie running. We found the current actually declines slightly when disk I/O is going because the cpu spends some fraction in I/O wait and doesn't run as hot. The five nodes below the line represent the majority of the worker nodes currently installed in GCC and were not otherwise part of the evaluation.

Vendor	Processor	Idle Amps	Loaded Amps
HP	Opteron 244	1.54	1.74
Rackable	Opteron 246HE	1.44	1.61
IBM	Opteron 246	1.55	1.80
Penguin	Opteron 248	1.70	1.98
ASA	Opteron 248	1.91	2.20
CSI	Opteron 248	1.95	2.30
Sun	Opteron 248	2.00	2.35
Koi	Opteron 250	2.07	2.44
Dell	Xeon 3.6/800	1.66	2.75
Koi	Xeon 3.6/800	1.70	3.10
Koi	Xeon 3.4/800	1.60	2.90
Koi	Xeon 3.0/800	1.20	2.70
Koi	Xeon 3.06/533	1.10	2.30
Koi	Xeon 2.66/533	1.00	1.90

Table 2: *Power consumption results.*

The power draw varies significantly among the four Opteron 248 hardware designs. This can be explained by the different configurations of power supply, fans, and disk drives in these units. The Penguin unit, most economical of the four, is cooled with a number of 1U fans and has a 350W power supply. The others come with power supplies of 400W or greater, a number of squirrel-cage blower fans, and in the case of the Sun, 15K RPM SCSI drives. For example the HP unit is well equipped with fans and power supplies and therefore draws proportionally more current for its speed, almost as much as the IBM unit.

The model that emerges for power consumption is that power consumption at a given CPU speed may vary by as much as 20% upwards depending on how many fans are added, and that each increment of Opteron CPU speed adds between 0.2 and 0.25A to the current draw.

For purposes of this study we will use 2.45A as the power draw for Opteron 250 (based on actual number of the only one we have), 2.2A as power draw for Opteron 248 (the average of the four we saw, with deviations of +/- 0.2A in either direction), 1.95A as power draw for Opteron 246 (the actual Opteron 246 we had is only 1.8A but it is likely that the Sun and ASA models would draw more than that if scaled down in processor speed), and 1.7A as power draw for Opteron 244. The trend is clear that Opterons will continue increasing in current draw as higher speeds are introduced, and the current phenomenon of modest current draw is a temporary condition, as is the current discrepancy between AMD and Intel hardware.

Given the stated goal of having on average 10 kW (kVA) per rack in the south room of GCC, this translates to an average goal of 2.1A current draw, or 250W, per node. Some of the Opteron 248 and 246 machines achieve this goal. None of the Xeon machines we can buy at the moment are that power-efficient. However, the Dell number above shows that it is possible to build a 3.6GHz unit that is somewhat less power-hungry than the Supermicro-designed case and power supply combination that we have been buying from Koi. We should also note that the recently released Pentium 4 600 uses 15% less power than its predecessors, so there is hope that the Intel Xeon dual processors will eventually see these same kind of innovations as well.

The Opteron 246HE is consumes significantly less power (7% less in idle 11% less under load) than its full voltage cousin. Unfortunately, AMD has only released the HE series in the 246 (2.0 GHz) frequency version. It also may have a slight price premium, although this price premium has dropped significantly since we ordered the unit. Rackable is the only one of the vendors who offered this solution to us. In addition, they offer rack-mounted AC to DC converters so each node would have a DC power supply, resulting in more reduction in power consumption and in heat generated inside each node.

It would be ideal to assign a power draw bonus, or penalty to each manufacturer in the bid. The logical thing to do would be to assign a value based on the fraction of the GCC power and cooling that one rack would use up. The Xeon 3.0/800 racks from Koi that were recently deployed use 12.9 kW, or 1.8% of the capacity per rack, assuming we have cooling for 72 racks at 10kW average each. The Opteron 246HE based machines would use just 7.7kW per rack, or 1.0% of the capacity per rack.

Taking for a rough number that the current south room of GCC cost $3 million to outfit and build the building, heating, and cooling infrastructure, 1% of this would be $30,000, 1.8% would be $54,000. It follows that every kilowatt saved in power saves us about $4000 of building infrastructure. The implications of how to evaluate price and performance in the light of infrastructure costs are discussed in the conclusions.

Unfortunately, it is difficult to predict what the power consumption numbers would be if we buy a machine of different speed than what was specified in the evaluation. Past experience has also told us that vendors can't be trusted to measure their power consumption accurately. Nor can AMD's numbers be trusted as they show the same thermal design power for several generations of the Opteron 2xx series even though we have measured them to run at different speeds. The model mentioned above is the best we can do at the moment.

6 Benchmark Applications

6.1 CMS

We wanted to run benchmarks related to CMS physics. Porting and compiling all of the CMS OO (namely OSCAR and ORCA) ([2]) software for the new platform to run in **(64Bit/64Bit)** mode requires a lot of work and the recompilation of all the underlying libraries (e.g. POOL, SEAL....). Therefore we benchmarked these applications only in **(32Bit/32Bit)** and **(64Bit/32Bit)** mode. We were specifically interested in the compatibility **(64Bit/32Bit)** option which allows to run a 64 Bit Operating system while still being able run all CMS software. The CMS OO applications were installed using **DAR** ([3]) which provides a 'nearly' complete runtime environment for various

CMS applications but doesn't allow for recompilation.

To demonstrate the gain in speed that one gets making full use of the new architecture, we also included several ROOT ([4]) based tests and also the Monte Carlo generator Pythia ([5]) in the test suite. CMS code and ROOT are dynamically linked C++ programs while PYTHIA is a statically linked FORTRAN program. This allowed us to run this application in all three operating modes.

The Applications needed to be compiled in 64 Bit. For the recompilation we used the g++ (ROOT) and g77 (Pythia) compilers based on gcc version 3.2.3 that is provided with Scientific Linux 3.03. We are using the 4.02/00 production version of ROOT and configured it for 64 Bit using the command:

configure linuxx8664gcc

- **ROOT:**

  ROOT provides various benchmarking routines testing various aspects with its distribution:

  - **bench:** this test program compares the I/O performance obtained with all STL collections of objects or pointers to objects and also Root collection class TClonesArray.
  - **stressLinear:** The suite of programs tests many elements of the vector, matrix and matrix decomposition classes (matrix size 100x100).
  - **stressgeom:** tests the ROOT geometry classes (TGeo).
  - **stress:** The suite of programs tests the essential parts of Root. In particular, there is an extensive test of the I/O system and Trees.

- **Pythia:** For the benchmark we run 100.000 super symmetry events at $\sqrt{s}$ =14 TeV (the Pythia main65.f example). The code is compiled with g77 with the O2 option:
  g77 -O2 -o main65 main65.f pythia6227.o

- **OSCAR_3_7_0:** The GEANT 4 based CMS detector simulation program. Here we simulate 300 single pion events of 50 GeV/c P_t.

- **ORCA_8_7_1:** The OSCAR output events are then digitized (writeAllDigis) and reconstructed (writeDST) with the corresponding ORCA applications:

  - **writeAllDigis**
  - **writeDST**

Table 6.1 shows the matrix of the different applications that we used to benchmark the performance and the different operation modes that we tested. Of course on the XEON 32 Bit machines that we used for comparison only the 32 Bit mode is available.

6.2 General Benchmarks

For this we used the following benchmarks: Tiny, a small track-reconstruction program which is able to run mostly in cache. Tiny has been used at Fermilab since well before the linux era and gives a measure of CPU speed in VAX-equivalents (VUPS). We also ran the CERN unit benchmark, and seti@home.

6.3 LQCD

The QCD applications are very dependent on memory bandwidth. We ran Streams, a standard memory bandwidth benchmark in both 32-bit and 64-bit mode, We also ran QCDStreamV, which adds measurement of the rate of floats and long doubles, as well as an estimate of MFlop/sec on multiplication of 3x3 complex matrices, the core calculation of lattice QCD. Finally, we ran the MILC lattice calculation in a variety of configurations.

6.4 Run II

Two Run II benchmarks were used. CDF supplied a self-contained tarball reconstruction executable based on ProductionExe5.3.1, 288 events. D0 supplied a reconstruction executable based on p14.06.01 plus 1000 events of high luminosity data. Both of these are new executables and thus the results will not be comparable with previous data on legacy machines.

Application	Operating mode:		
	32-bit legacy (32Bit/32Bit)	Compatibility (64Bit/32Bit)	64-bit (64Bit/64Bit)
ROOT: stress	Yes	Yes	Yes
ROOT: stress × 2	Yes	Yes	Yes
ROOT: stressgeom	Yes	Yes	Yes
ROOT: stressLinear	Yes	Yes	Yes
ROOT: bench	Yes	Yes	Yes
Pythia	Yes	Yes	Yes
Pythia × 2	Yes	Yes	Yes
OSCAR	Yes	Yes	N/A
OSCAR × 2	Yes	Yes	N/A
ORCA Digitization	Yes	Yes	N/A
ORCA Digitization × 2	Yes	Yes	N/A
ORCA Reconstruction	Yes	Yes	N/A
ORCA Reconstruction × 2	Yes	Yes	N/A

Table 3: *Matrix of various benchmarks run in the different operating modes. The ×2 indicates that both processors were running the application simultaneously.*

6.5 SDSS

The Sloan Digital Sky Survey is planning a Supernova Data Pipeline to enable a rapid turnaround for supernova searches. They ran a combination of the SDSS Photo package for astrometric/photometric reduction, and the frame subtraction code which finds supernova candidates by comparing the search image to previously recorded template images of the same piece of sky. The goal is to be able to process one night's data before the next night's observation.

7 Benchmarking results

7.1 CMS

Pythia Benchmarking results						
	Operating mode:					
	32-bit legacy (32Bit/32Bit)		Compatibility (64Bit/32Bit)		64-bit (64Bit/64Bit)	
CPU	single (Events/sec)	× 2 (average) (Events/sec)	single (Events/sec)	× 2 (average) (Events/sec)	single (Events/sec)	× 2 (average) (Events/sec)
244	91.0	91.0	90.3	90.4	110.1	110.0
246	101.3	101.4	100.9	101.1	121.2	121.5
246HE	101.1	101.2	100.5	100.6	122.4	122.5
248	113.0	112.9	112.3	112.3	136.4	136.1
250	121.3	121.3	120.8	120.7	146.9	146.8
XEON 3.4	88.5	88.7	88.1	88.3	108.6	108.6
XEON 3.6	93.5	94.21	93.4	93.4	115.6	115.4
XEON 2.4	65.5	65.5	-	-	-	-
XEON 2.8	75.9	75.9	-	-	-	-
XEON 3.06	86.8	75.7	-	-	-	-
PIII 1.0	40.3	40.3	-	-	-	-

Table 4: *Summary of Pythia Benchmarking results.*

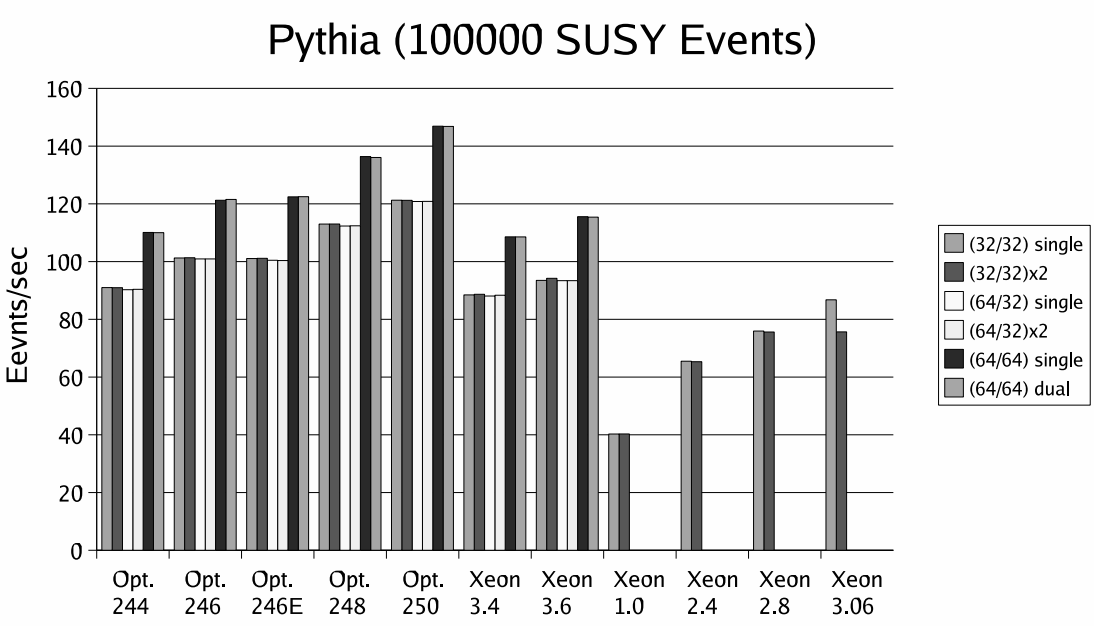

Figure 3: *Summary of Pythia Benchmarking results.*

ROOT Stress						
	Operating mode:					
	32-bit legacy (32Bit/32Bit)		Compatibility (64Bit/32Bit)		64-bit (64Bit/64Bit)	
CPU	single ROOTMarks	× 2 (average) ROOTMarks	single ROOTMarks	× 2 (average) ROOTMarks	single ROOTMarks 8	× 2 (average) ROOTMarks
244	723.6	715.35	672.5	673.3	1059.9	1050.75
246	797.1	793.45	751.6	748.25	1176.1	1165.65
246HE	801	805.05	752.2	749.1	1183.9	1170.35
248	894.2	888.4	834.7	835.75	1320.8	1305.95
250	957.6	956.7	898.3	891.35	1402.6	1395.55
XEON 3.4	942.1	830.85	909.3	903.8	1145.9	1135.8
XEON 3.6	1034.8	1020.6	969.7	958.7	1208.6	1194.4
XEON 2.4	590.7	583.75	-	-	-	-
XEON 2.8	731	720.05	-	-	-	-
XEON 3.06	791.6	626.0	-	-	-	-
PIII 1.0	344	332	-	-	-	-

Table 5: *Summary of Root Stress Benchmarking results. We observe that the application runs about 6% slower in compatibility mode compared to (32/32). Also in this case the Opteron runs more than 40 % faster in 64Bit mode compared to (32/32) while the 64 Bit Xeons only gain about 20% .*

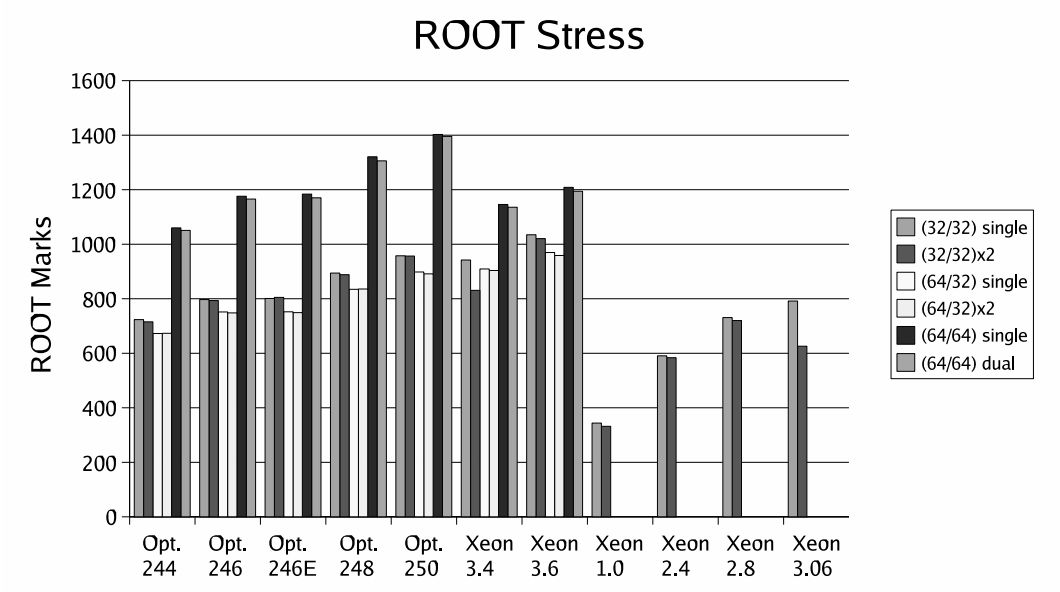

Figure 4: *Summary of Root Stress Benchmarking results. We observe that the application runs about 6% slower in compatibility mode compared to (32/32) mode. Also in this case the Opteron runs more than 40 % faster in 64Bit mode compared to (32/32) while the 64 Bit Xeons only gain about 20% . Note the Opteron 250 in (64/64) Bit mode runs about 2.4 times faster than our current worker nodes (2.4 GHz Xeons in (32/32)).*

ROOT Benchmarks									
	Operating mode:								
	32-bit legacy (32Bit/32Bit)			Compatibility (64Bit/32Bit)			64-bit (64Bit/64Bit)		
CPU	geom	linear	bench	geom	linear	bench	geom	linear	bench
244	673	626.1	798.59	657.4	589.9	617.15	1210.9	1017.2	1035.76
246	739.1	694.7	877.73	731	653.7	684.86	1323.6	1121.4	1160.47
246HE	737.8	697.4	883.65	735.1	653.5	681.15	1350.5	1129.6	1147.12
248	826.6	766.9	978.33	814.7	726	758.83	1497.7	1250.7	1277.26
250	893.3	833	1049.86	875.6	783.2	815.84	1632.8	1353.1	1374.24
XEON 3.4	757.4	849.1	1078.74	746.1	807.3	917.49	1293.5	1202.4	1190.89
XEON 3.6	796.8	905.7	1151.87	798.4	863.4	975.32	1369.1	1274.4	1265.8
XEON 2.4	575.7	599.3	603.79	-	-	-	-	-	
XEON 2.8	692.9	746.1	841.28	-	-	-	-	-	
XEON 3.06	727	812.5	907.69	-	-	-	-	-	
PIII 1.0	365.5	342.5	360.81	-	-	-	-	-	

Table 6: *Summary of various Root Benchmarking results.*

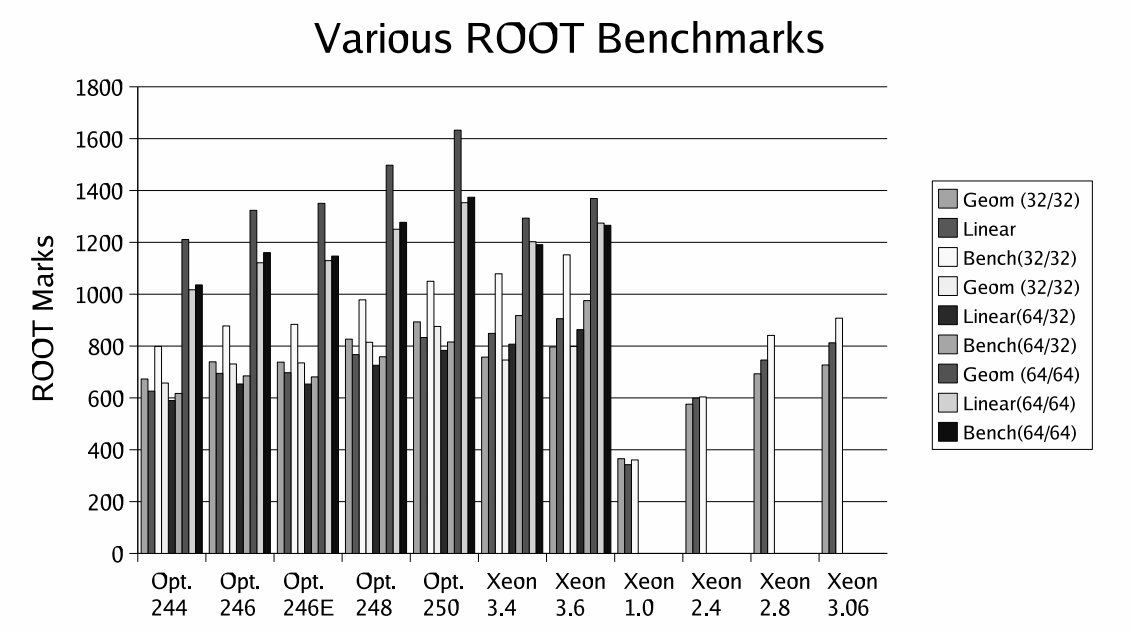

Figure 5: *Summary of various ROOT Benchmarking results.*

CMS OO Benchmarks									
	Operating mode:								
	32-bit legacy (32Bit/32Bit)			Compatibility (64Bit/32Bit)					
CPU	OSCAR single (Evt./sec)	OSCAR ×2 (av.) (Evt./sec)	DIGIS single (Evt./sec)	OSCAR single (Evt./sec)	OSCAR ×2 (av.) (Evt./sec)	DIGIS single (Evt./sec)	DIGIS ×2 (av.) (Evt./sec)	DST single (Evt./sec)	DST ×2 (av.) (Evt./sec)
244	0.0915	0.0923	0.1171	0.0866	0.0867	0.0989	0.0980	0.809	0.806
246	0.1021	0.1028	0.1293	0.0956	0.0959	0.1092	0.108	0.901	0.895
246HE	0.1037	0.1026	0.1325	0.0967	0.0966	0.1103	0.1104	0.912	0.898
248	0.1147	0.1139	0.1430	0.1067	0.1074	0.1210	0.1196	0.974	0.988
250	0.1186	0.1221	0.1562	0.1154	0.1156	0.1300	0.1296	1.039	1.035
XEON 3.4	0.1067	0.1060	0.1403	0.0938	0.0929	0.1251	0.1238	0.929	0.890
XEON 3.6	0.1066	0.1057	0.1427	0.1012	0.1012	0.1332	0.1310	1.026	1.019
XEON 2.4	0.0603	0.0604	0.0868	-	-				
XEON 2.8	0.0715	0.0702	0.0996	-	-				
XEON 3.06	0.0790	0.0637	0.1078	-	-				
PIII 1.0	0.0343	0.0330	0.0501	-	-				

Table 7: *Summary of CMS Benchmarking results.*

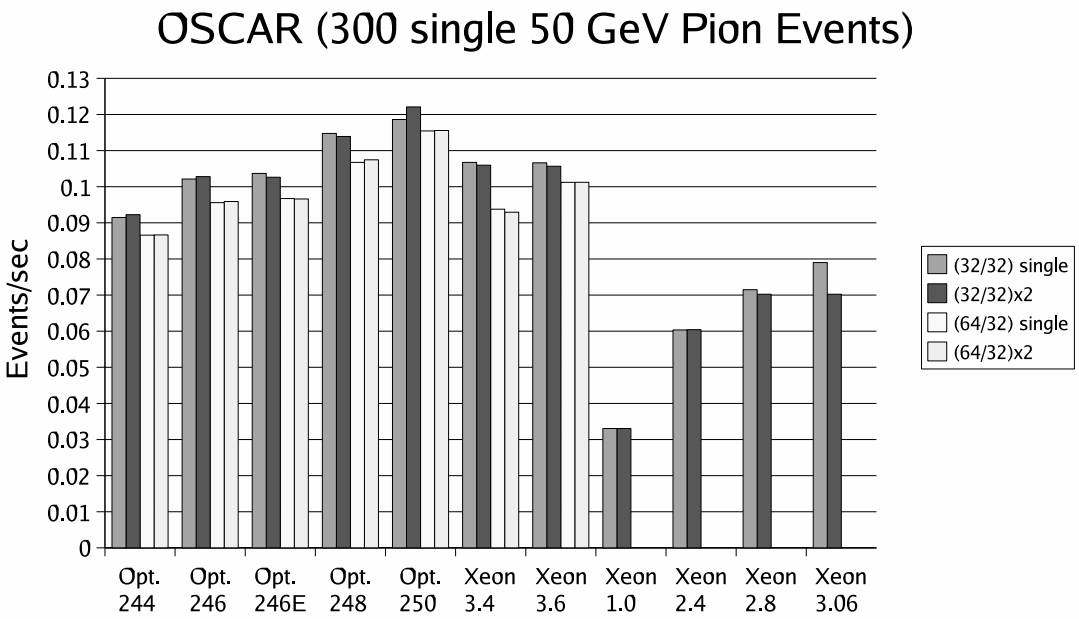

Figure 6: *Summary of OSCAR Benchmarking results. Again as for the Root stress results we observe that the application runs about 5 % slower in compatibility mode.*

7.2 General Benchmarks

We ran the "tiny" benchmark on all machines. We first used the legacy executable that has been used throughout the farms era at Fermilab, non-optimized in 32-bit mode, then tried with other compilers. On both platforms we used the gcc compiler with optimization. For the Opterons we also used the Pathscale compiler, and for Xeons we used the Intel compiler. All gcc and Pathscale/Intel runs were done in 64-bit mode using 64-bit binaries.

The Intel 8.1 version compilers generate SSE3 instructions as part of their optimized code, which are incompatible with current Opterons, so any use of the Intel 8.1 compiler will not use the Opteron's instruction set to its full potential. Nevertheless, there are still some compiler switches that can be used. In the Cern unit benchmarks below there is still some gain to be had by using the Intel 8.1 compiler on the Opterons. The one Xeon number is compiled using the similar non-optimal switches for comparison.

The seti@home results were compiled using 32-bit seti binaries in 32-bit mode. A seti binary is available optimized for x86_64 but we did not use it. We use seti@home software version 3.03 with a fixed work unit captured from the seti site

Processor	VUPS Legacy	VUPS gcc	VUPS proprietary
Opteron 244	966	1813	1988
Opteron 246	1076	2027	2223
Opteron 246HE	1076	2017	2212
Opteron 248	1209	2230	2451
Opteron 250	1290	2440	2677
Xeon 3.6	1386	2309	2910
Xeon 3.4	1307		
PIII 1GHz	568		

Table 8: *Tiny results.*

Processor	Cern unit gcc	Cern unit IFC 8
Opteron 244	902	1377
Opteron 246	1002	1540
Opteron 246HE	1004	1531
Opteron 248	1119	1726
Opteron 250	1200	1837
Xeon 3.6	1024	1116

Table 9: *Cern Unit results.*

Processor	seti time (sec)
Opteron 244	9809
Opteron 246	8687
Opteron 246HE	8701
Opteron 248	7721
Opteron 250	7332
Xeon 3.6	6863
Xeon 3.4	7647

Table 10: *seti@home results (lower is better)*

7.3 LQCD

The LQCD benchmarks were run only on the eight Opteron nodes and one reference Xeon node. Thirteen trials were run, they are summarized one by one.

The STREAM benchmark is a simple synthetic benchmark program that measures sustainable memory bandwidth (in MB/s) and the corresponding computation rate for simple vector kernels. The following plot is for a single stream process. The stream binary was compiled with gcc on a 32-bit linux kernel.

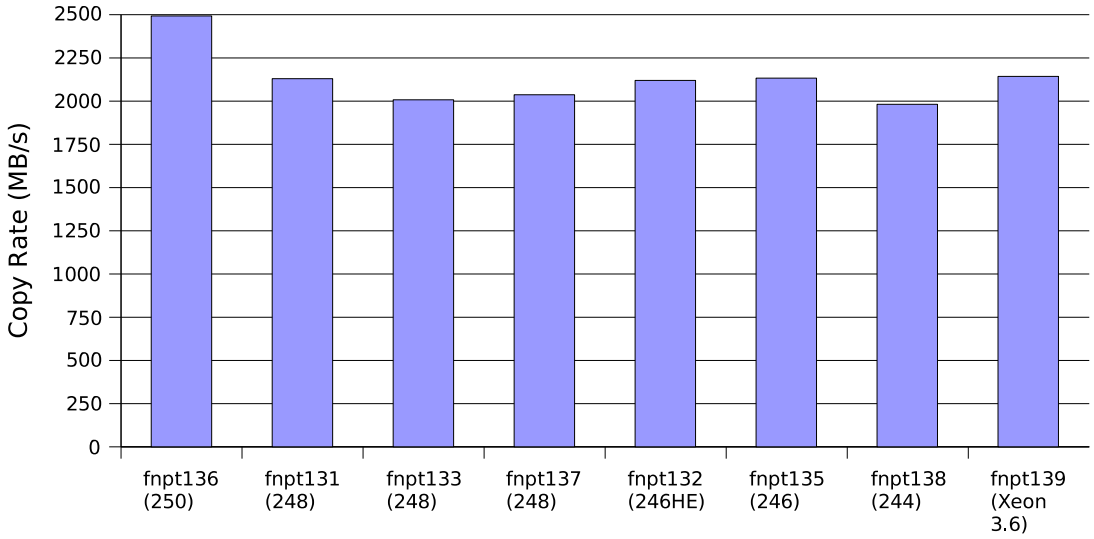

Figure 7: *Summary of single process stream benchmark under 32-bit Linux. Higher is better*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)	fnpt139 (Xeon3.6)
2493	2130	2108	2037	2120	2133	1982	2143

The following plot is for a single stream process. The stream binary was compiled with gcc on a 64-bit (x86_64) Linux kernel.

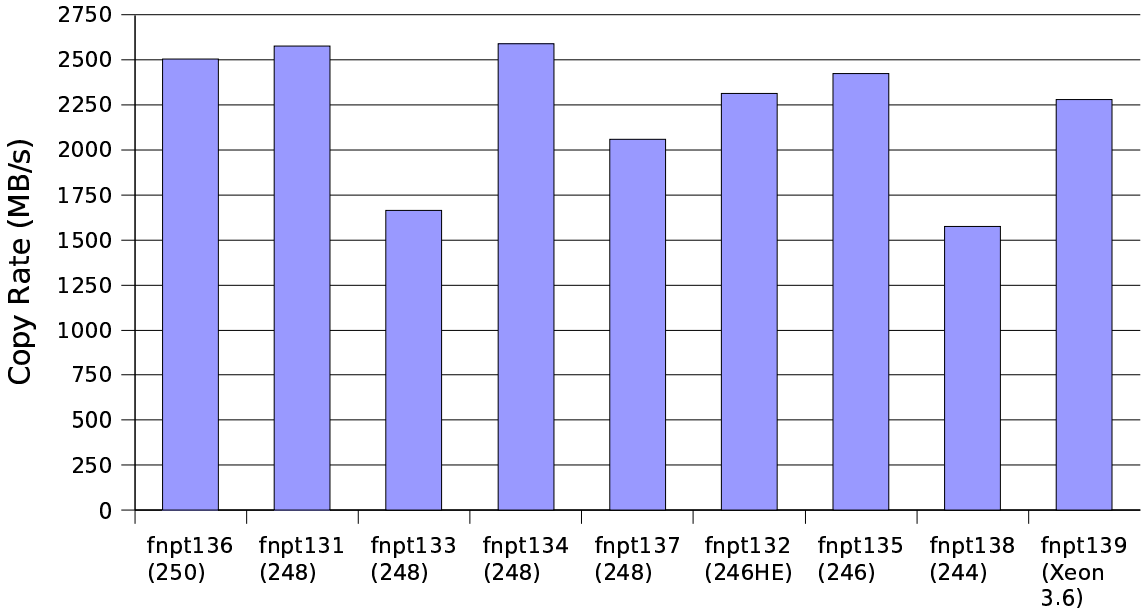

Figure 8: *Summary of single process stream benchmark under 64-bit Linux. Higher is better*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt134 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)	fnpt139 (Xeon3.6)
2507	2576	1663	2592	2058	2315	2427	1575	2280

In the STREAM ”Copy” measurement, the rate of movement of double precision values between arrays is measured. QCDStreamV adds measurements of the rate of movement using floats and long doubles. QCDStreamV calculates sustained MFlop/sec during matrix-vector multiplies. The matrices are 3x3 complex, and the vectors are 3x1 complex. The following plots are from a single QCDStreamV process where matrices and vectors are pulled sequentially from the arrays. The QCDStreamV binary was compiled with inline gcc assembler on a 32-bit Linux kernel.

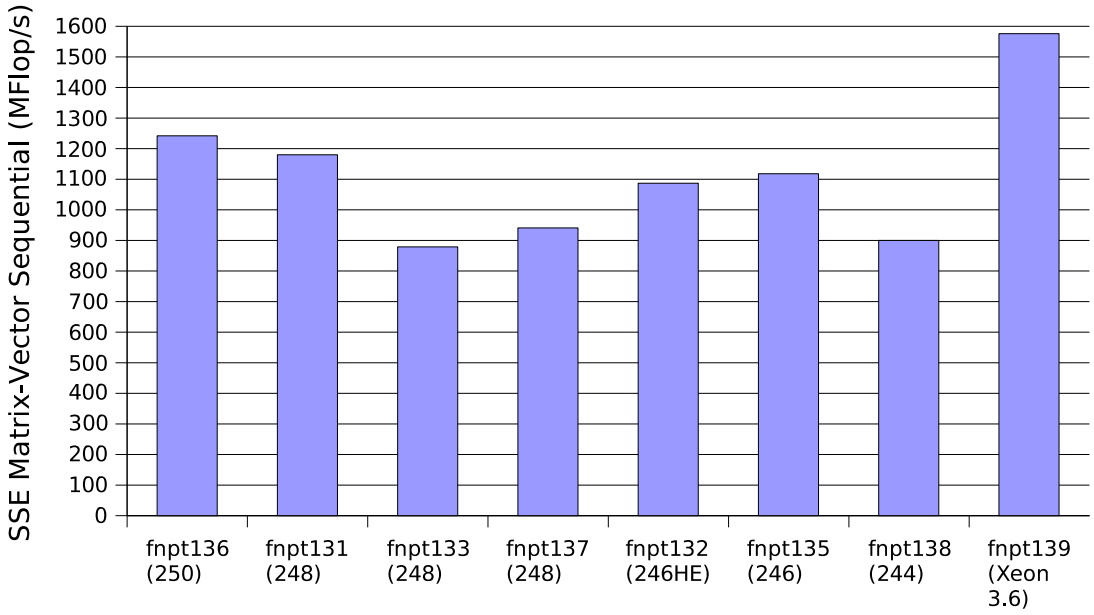

Figure 9: *Summary of single process QCDStreamV benchmark under 64-bit Linux. Higher is better*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)	fnpt139 (Xeon3.6)
1242	1180	879	941	1087	1118	900	1576

The following plot is from a single QCDStreamV process where matrices and vectors are pulled sequentially from the arrays. The QCDStreamV binary was compiled with inline gcc assembler on a 64-bit (x86_64) Linux kernel.

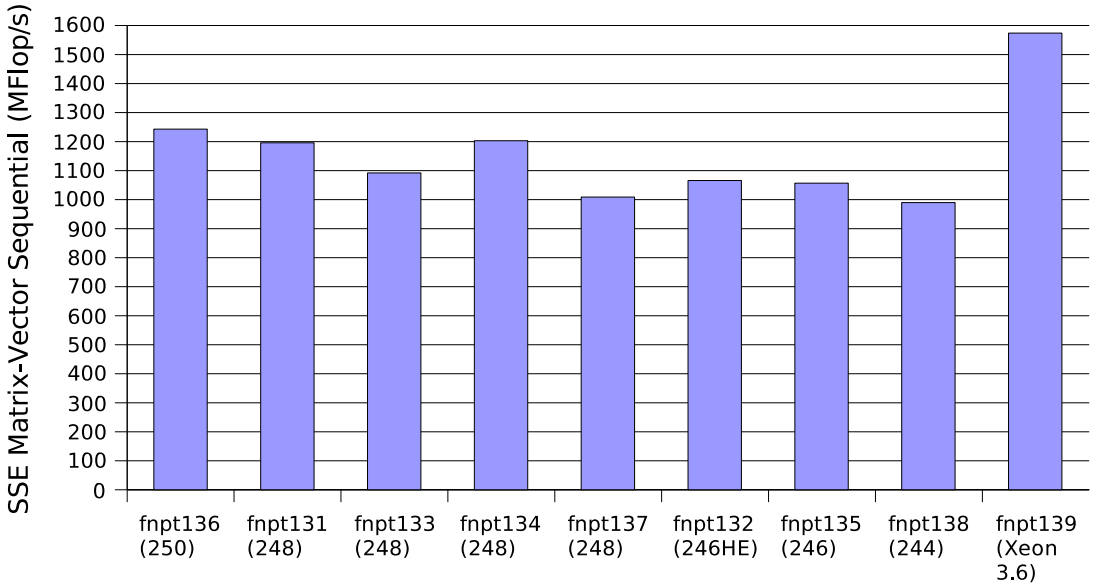

Figure 10: *Summary of single process QCDStreamV benchmark under 64-bit Linux. Higher is better*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt134 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)	fnpt139 (Xeon3.6)
1243	1196	1092	1203	1009	1066	1057	990	1574

The Lattice QCD MILC benchmark measures available memory bandwidth and CPU cache performance. All the lattice QCD MILC benchmarks listed here are for single-precision Kogut-Susskind conjugate gradient. The following Lattice QCD MILC benchmark was run as a single process. The code was compiled with gcc on a 32-bit Linux kernel. The lattice size used for the run was 10.8MBytes.

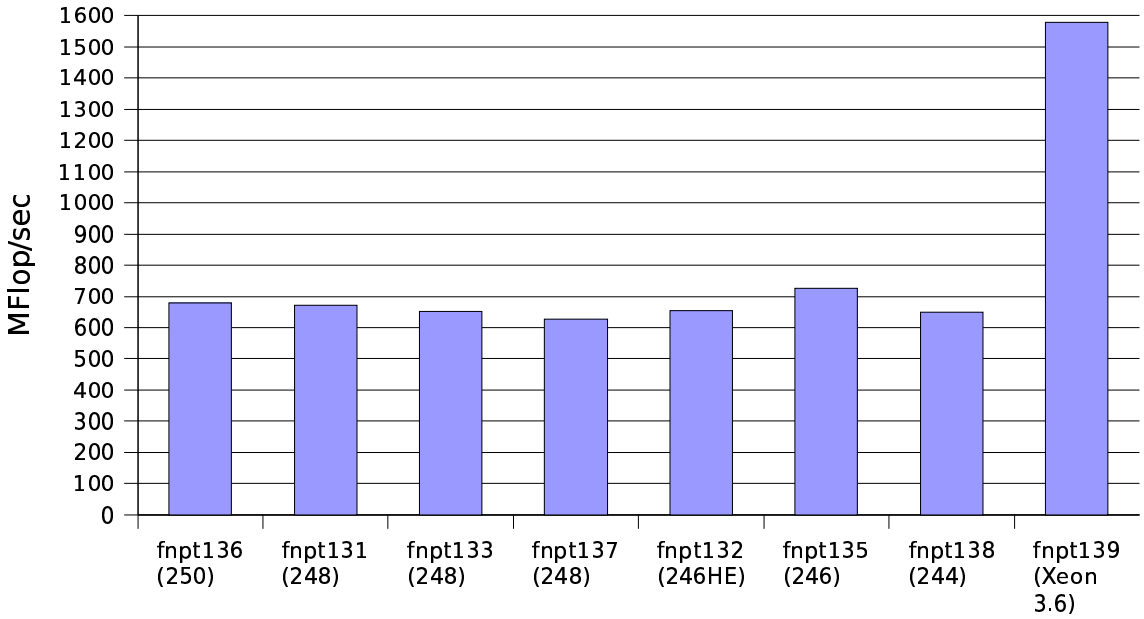

Figure 11: *Summary of single process QCD MILC benchmark under 32-bit Linux. Higher is better*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)	fnpt139 (Xeon3.6)
678	672	653	628	655	726	649	1579

The following Lattice QCD MILC benchmark was run as a two-node mpi process. The mpi version from ANL was compiled to use shared memory. The code was compiled with gcc on a 32-bit Linux kernel. The lattice size was 11.2 MBytes.

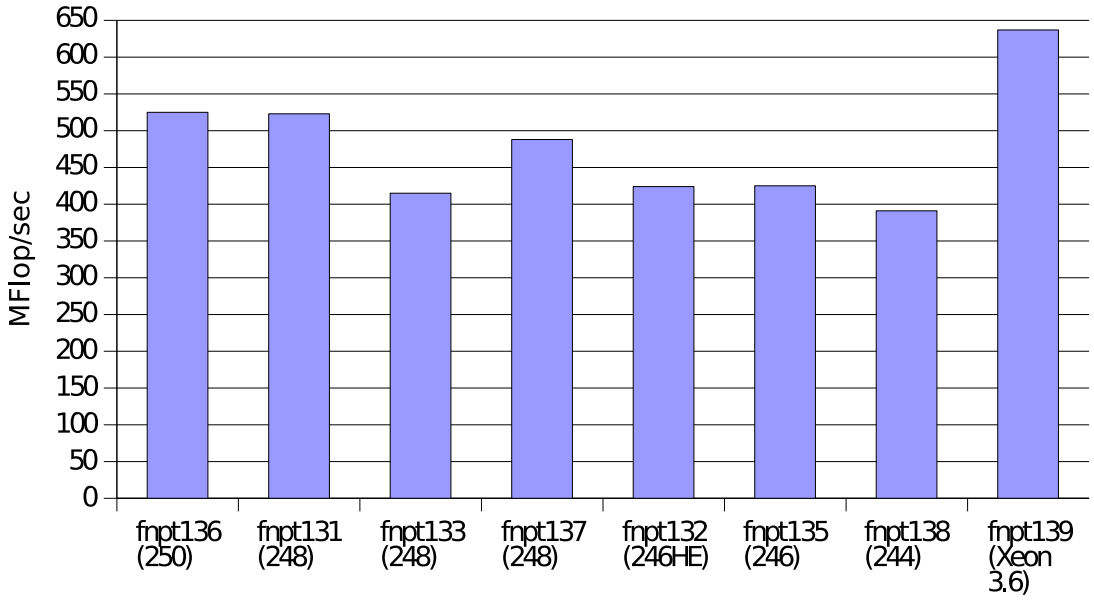

Figure 12: *Summary of QCD MILC benchmark run as a two node mpi process under 32-bit Linux. Higher is better.*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)	fnpt139 (Xeon3.6)
525	523	415	488	424	425	391	637

The following Lattice QCD MILC benchmark was run as a single process. The code was compiled using gcc with QDP optimizations on a 32-bit Linux kernel. The lattice size was 10.8 MBytes

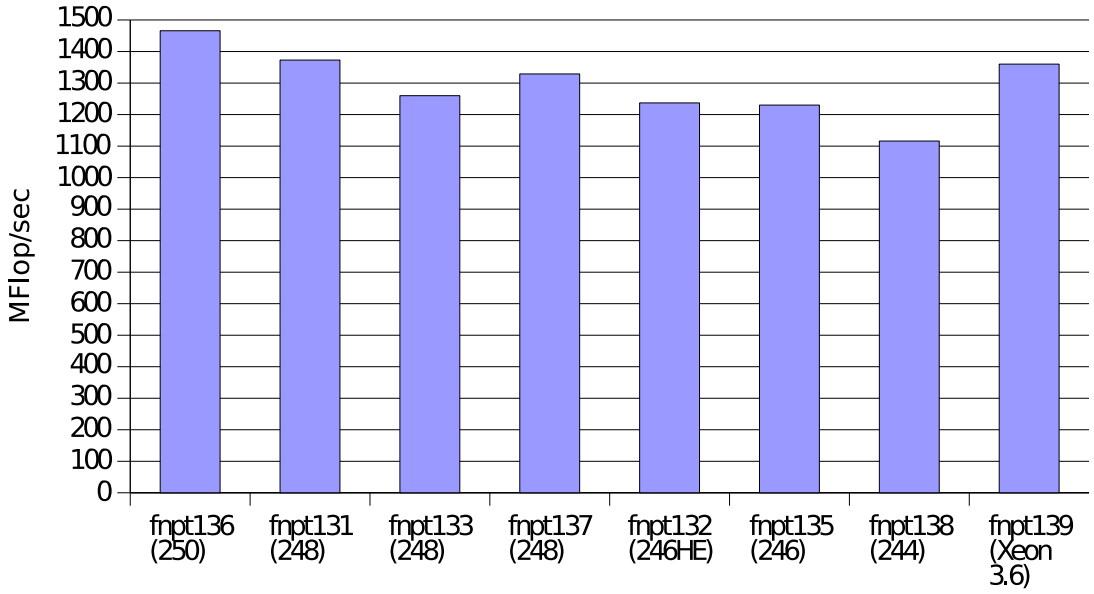

Figure 13: *Summary of QCD MILC benchmark compiled with QDP optimizations, under 32-bit Linux. Higher is better.*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)	fnpt139 (Xeon3.6)
1466	1373	1260	1329	1237	1230	1116	1360

The following Lattice QCD MILC code was run as a two node mpi process. The code was compiled using gcc with QDP optimizations on a 32-bit Linux kernel. The lattice size was 11.2 MBytes.

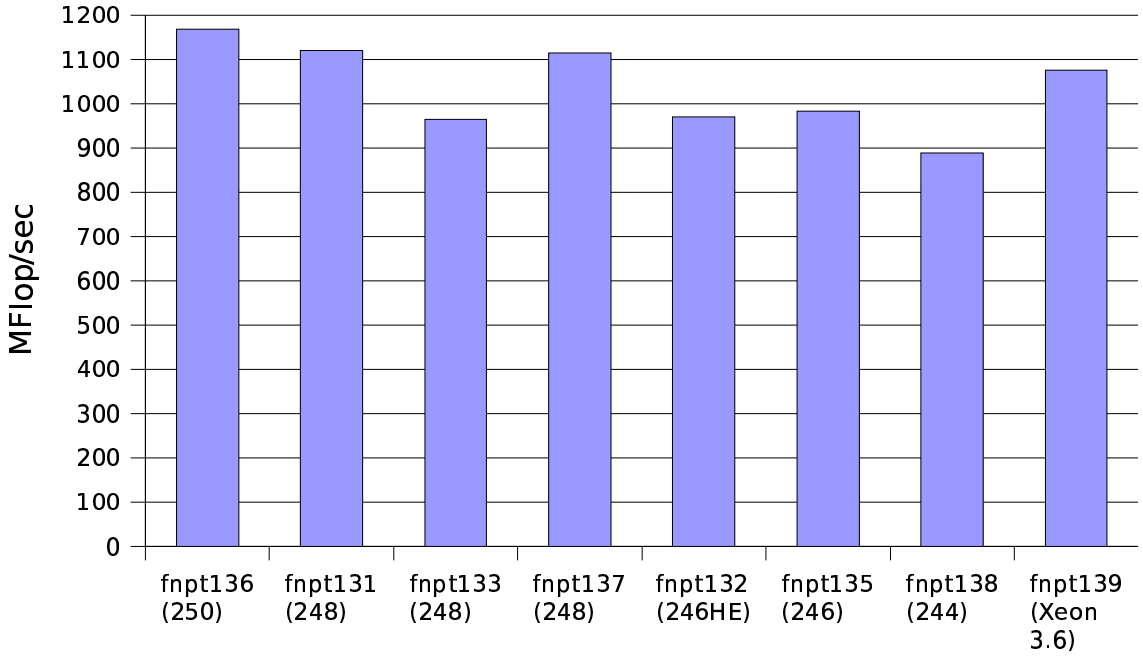

Figure 14: *Summary of QCD MILC benchmark, compiled with QDP optimizations and run as a two node mpi process, under 32-bit Linux. Higher is better.*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)	fnpt139 (Xeon3.6)
1169	1120	965	1115	970	984	889	1076

The following Lattice QCD MILC code was run as a single process. The code was compiled using gcc on a 64-bit (x86_64) Linux Kernel. The lattice size was 10.4 MBytes

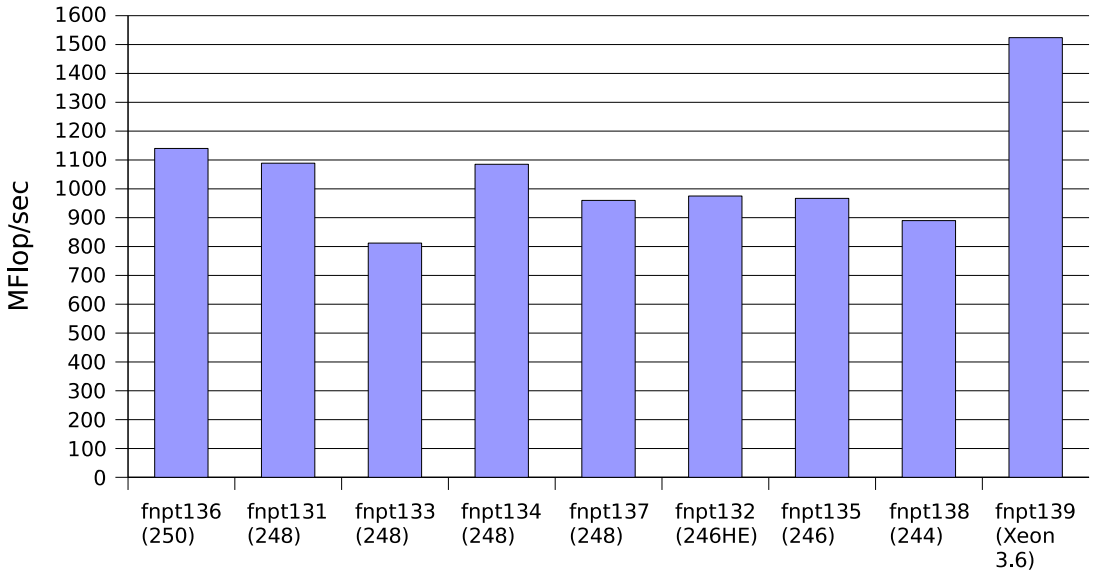

Figure 15: *Summary of single process QCD MILC benchmark under 64-bit Linux. Higher is better.*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt134 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)	fnpt139 (Xeon3.6)
1140	1089	812	1085	960	975	967	890	1524

The following Lattice QCD MILC code was run as a two node mpi process. The code was compiled using gcc on a 64-bit (x86_64) Linux Kernel. The lattice size was 10.2 MBytes

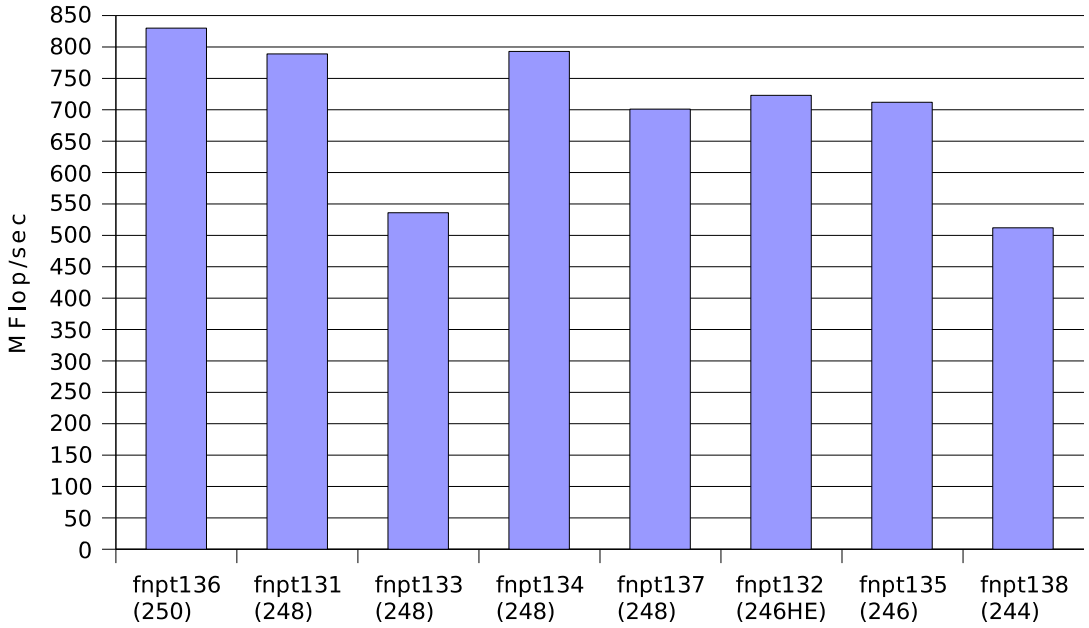

Figure 16: *Summary of QCD MILC benchmark run as a two node mpi process, under 64-bit Linux. Higher is better.*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt134 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)
830	789	536	793	701	723	712	512

The following Lattice QCD MILC code was run as a single process. The code was compiled using the PathScale compiler on 64-bit (x86_64) Linux kernel. The lattice size was 11.3 MBytes

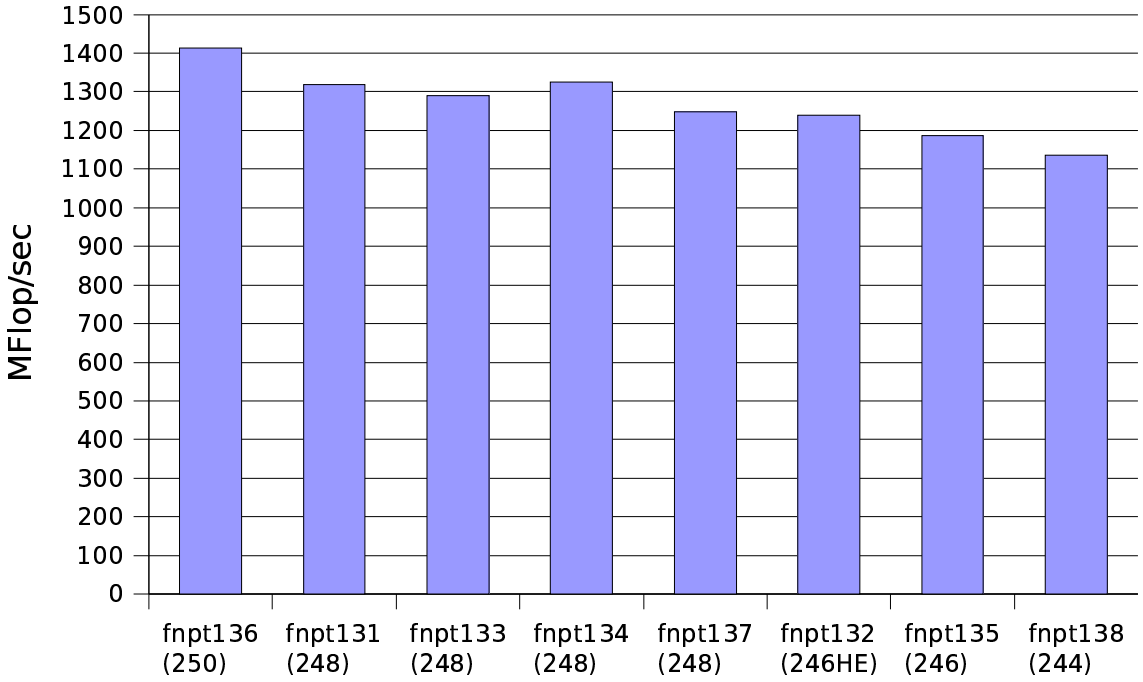

Figure 17: *Summary of single process QCD MILC benchmark compiled using the PathScale compiler under 64-bit linux. Higher is better.*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt134 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)
1414	1318	1291	1326	1249	1239	1186	1136

The following Lattice QCD MILC code was run as a two node mpi process. The code was compiled using the PathScale compiler on 64-bit (x86_64) Linux kernel. The lattice size was 11.7 MBytes

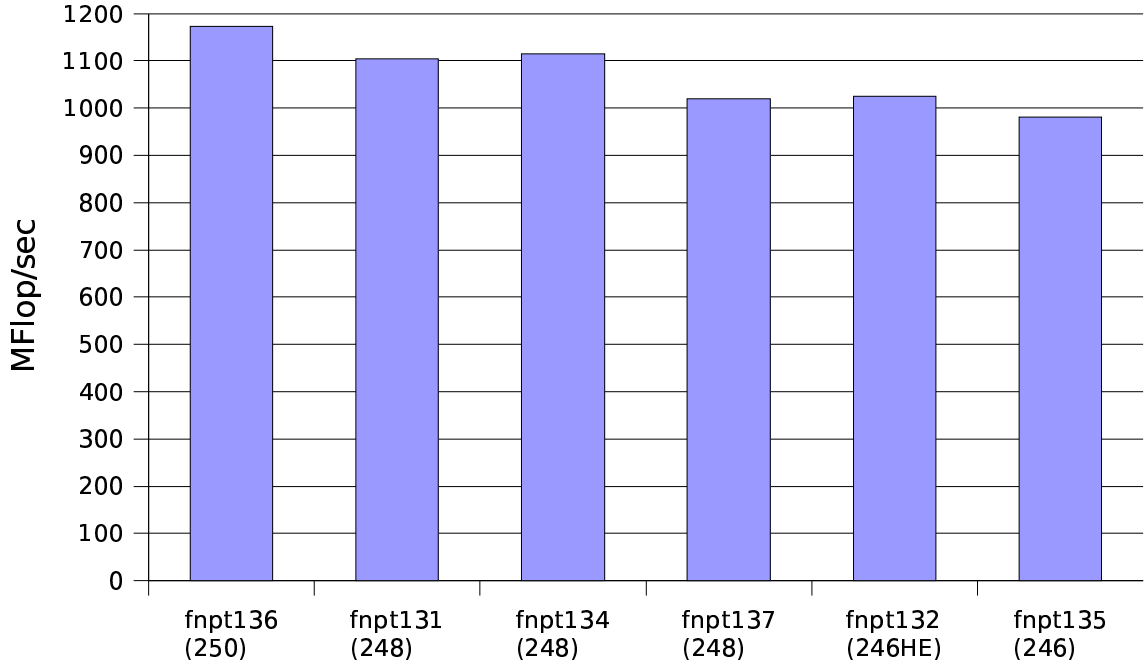

Figure 18: *Summary of QCD MILC benchmark compiled using the PathScale compiler and run as a two node mpi process, under 64-bit Linux. Higher is better.*

fnpt136 (250)	fnpt131 (248)	fnpt134 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)
1172	1104	1114	1019	1025	981

The following Lattice QCD MILC code was run as a single process. The code was compiled using the PathScale compiler on 64-bit (x86_64) Linux kernel. The lattice size was 11.3 MBytes

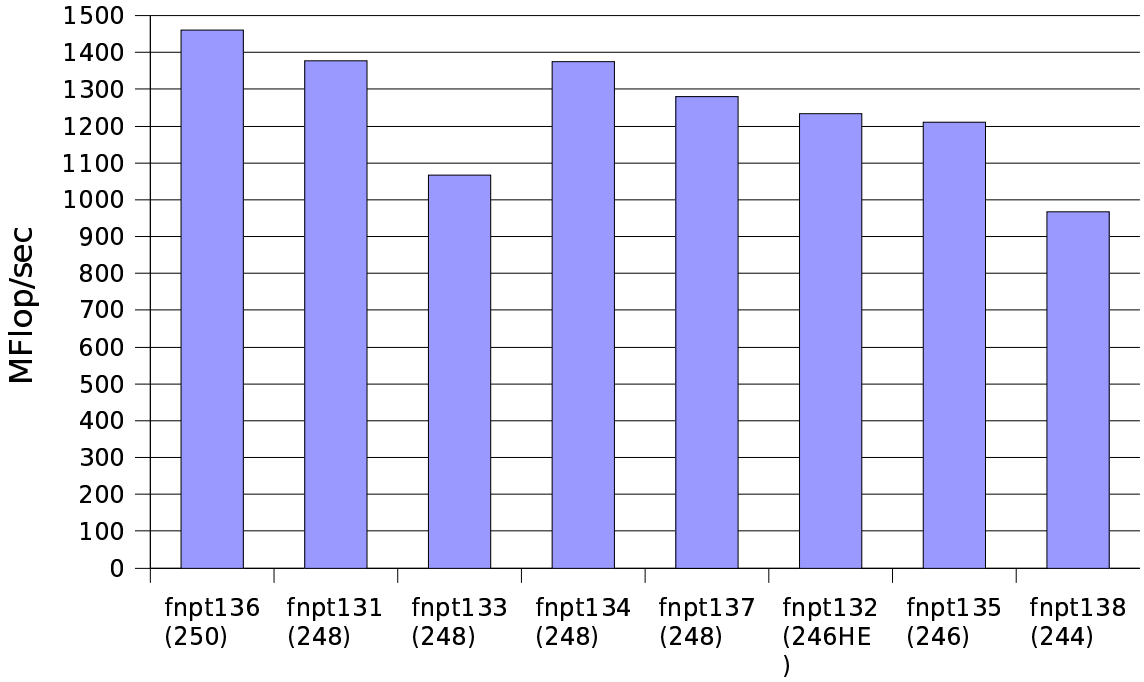

Figure 19: *Summary QCD MILC benchmark compiled with QDP optimizations, under 64-bit linux. Higher is better.*

fnpt136 (250)	fnpt131 (248)	fnpt133 (248)	fnpt134 (248)	fnpt137 (248)	fnpt132 (246HE)	fnpt135 (246)	fnpt138 (244)
1461	1377	1067	1375	1280	1234	1211	967

7.4 Run II

We present the results of the two benchmarks in events reconstructed per second. In addition we compare the performance with that of a Pentium III 1 GHz machine. 1000 Fermi Cycles is defined to be the performance of the Pentium III 1GHz machine on the average of CDF and D0 event reconstruction software. In this case, therefore, the Fermi cycles will be 1000 times the ratio.

Processor	D0 Evts/s	Rel PIII	CDF Evts/s	Rel PIII	Avg
Opteron 244	0.0187	2.28	1.05	2.56	2.42
Opteron 246	0.0206	2.51	1.12	2.73	2.62
Opteron 246HE	0.0208	2.54	1.11	2.71	2.62
Opteron 248	0.0230	2.80	1.24	3.02	2.91
Opteron 250	0.0240	2.93	1.33	3.24	3.09
Xeon 2.4	0.0132	1.61	0.82	2.00	1.80
Xeon 2.66	0.0146	1.78	0.90	2.20	1.99
Xeon 3.06	0.0164	2.00	1.05	2.56	2.28
Xeon 3.0	0.0162	1.98	1.00	2.44	2.21
Xeon 3.4	0.0183	2.23	1.12	2.73	2.48
Xeon 3.6	0.0190	2.32	1.20	2.93	2.62
Athlon 1.66	0.0134	1.63	0.79	1.92	1.78
PIII 1.0	0.0082	1.00	0.41	1.00	1.00

Table 11: *Run II benchmarks*

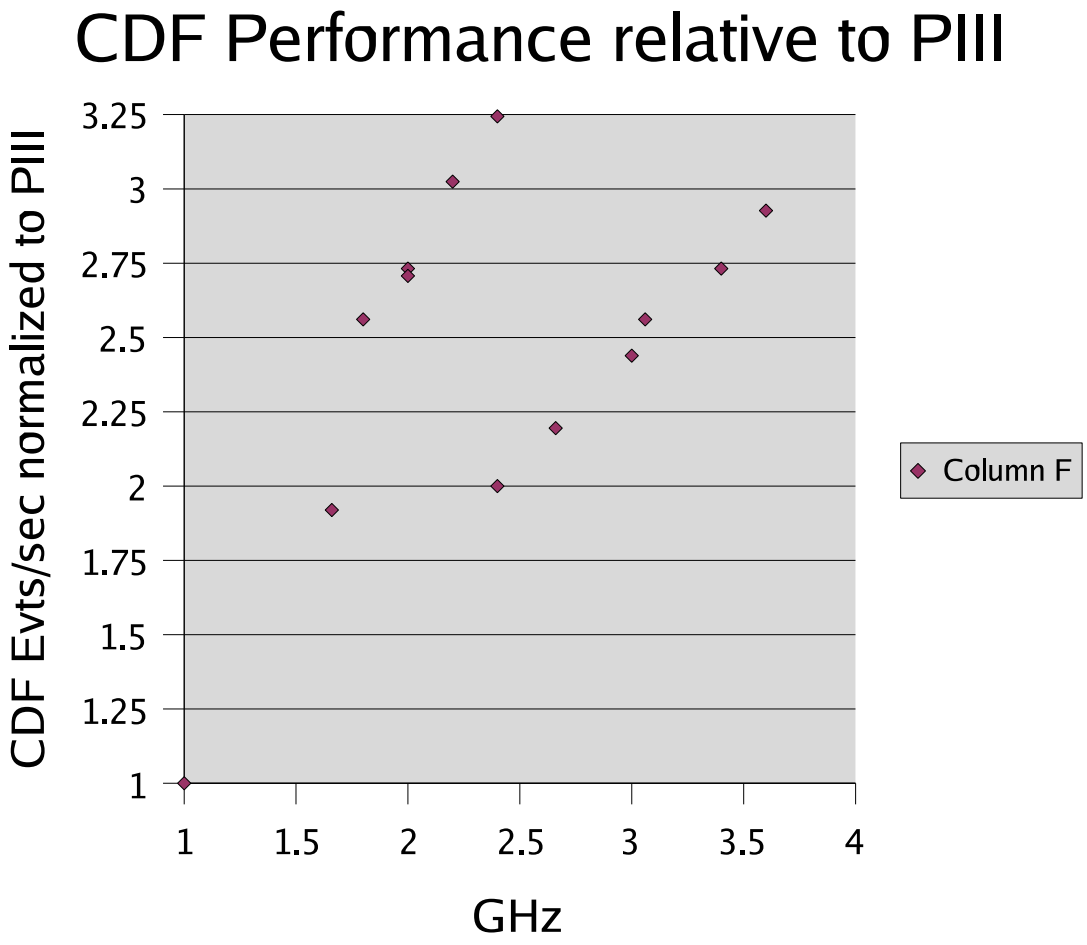

Figure 20: *CDF reconstruction relative to PIII 1 GHz.*

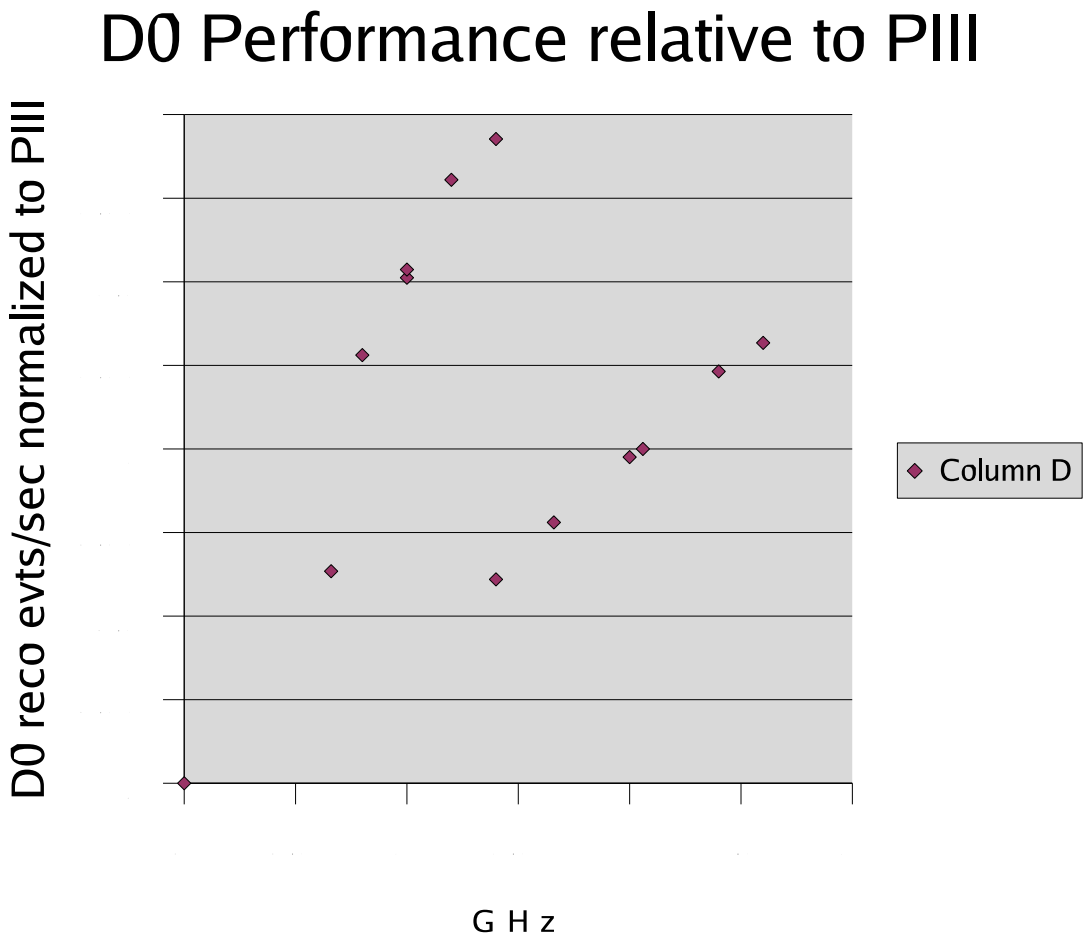

Figure 21: *D0 reconstruction relative to PIII 1 GHz.*

8 Conclusions

8.1 Benchmarks

8.1.1 CMS

CMS software runs in 64 Bit compatibility mode and 32 Bit native mode without any modifications to the code. We observe OSCAR and ROOT applications to run around 5% slower in compatibility mode than in legacy mode. The effect is even more pronounced for the digitization step where the loss in performance is around 15 %. When running in 64 Bit native mode we observe speed increases ranging from 21 % (Pythia) to 49 % (SGeom) compared to 32 Bit native mode. So it will be interesting to see how CMS code would perform after being ported to the new platform. Currently we definitely can't use all the performance provided by the new platform. The Opteron 250 that we tested in (64/64) Bit mode runs the ROOT stress test about 2.4 times faster than the current CMS worker nodes (2.4 GHz Xeons in (32/32)).

When porting the code to 64 Bit cernlib probably would have to be dropped since there are no plans to port it to 64 bit and the code assumes that the data types int, long and pointer have the same size which is not the case for AMD64/EM64T.

If we compare the similarly priced Opteron and Xeon chips, (Opteron 250 and Xeon 3.6) we see that in 64-bit applications in 64-bit mode, (Pythia and Root) the Opteron hardware has a clear advantage over the Xeon, given the compilation options that were used. The Opteron appears to be faster on the OSCAR application as well. In 32-bit Root applications the Xeon has a slight edge.

8.1.2 General Benchmarks

The Tiny results show above that for some applications, a good optimizing compiler can give a factor of two performance improvement on this hardware. Significant gains can be made with gcc, the proprietary compilers do even better. The profile feature of the Intel compilers can make the Intel hardware even faster on Tiny.

The numbers we show here on the Cern Unit benchmark show the Opterons to be superior but this would change if the full optimization features of the Intel 8.1 compiler were used on the Intel hardware.

seti@home is one application which appears to perform better on Xeon hardware than on the corresponding Opteron hardware. This may be a function of how the seti binary is compiled and optimized. The main result from seti, which is very memory-intensive, is that both for the Opteron and Xeon machines, the time that it takes to run two copies of seti on two processors is within 1% of the time it takes to run just one. This is a significant improvement from earlier hardware where the effect due to the crowded memory bus could be as much as 15% lengthening of the running time when both processors are running seti.

8.1.3 LQCD

The following conclusions can be drawn from LQCD on Opteron hardware: The QCDStreamV number of the Xeon 3.6 is considerably higher than any of the Opterons due to the optimizations that are available for the Xeons in the compilers, even GCC. The SSE instructions and the adaptive hardware prefetching give the Intel chip an advantage for this code. This is also the case for the single process QCD MILC code, which has hand-coding in assembler specifically for the Xeon architecture.

QDP optimizations that were mentioned in those graphs that have them are an attempt to put the same type of adaptive prefetching into the Opteron chip, helping the Opterons improve their performance significantly on the single-process MILC benchmarks, almost a factor of two.

However, when the MPI (message-passing) numbers are shown with two processes are going on the same machine, the Opterons do proportionately better. This is due to the NUMA construction of the machine in which each CPU can access its own memory and not have contention between two CPU's contending on the same memory for the memory bus.

Although dual Opterons do better than dual Xeons in the performance market, the LQCD group is likely to buy the single-processor equivalent of the Xeon, namely the Pentium 4 600 series chip, which has similar performance to the Xeon. This is for price-performance reasons.

8.1.4 Run II

The results we see with the legacy Run II executables, which are compiled for Pentium-III generation hardware, are similar to the effects that we saw on AMD Athlon hardware when it first came out. Legacy executables of the D0 experiments do better on Opteron hardware than they do on Xeon hardware, by a big factor (26%). We believe this is due to the implementation of legacy floating-point instructions, which tends to be more efficient in AMD hardware than in Netburst (P4/ Xeon) architecture.

On the Opteron machines we were forced to run the D0 benchmark in 64-bit compatibility mode as that was the only way we could make it run under any variant of Scientific Linux. It's likely, therefore, that the actual performance of the D0 executables will be better than the numbers we have here. The CMS benchmarks showed that such executables run slower in 64-bit compatibility mode compared to 32 bit native.

CDF benchmarks also are faster on Opterons than on Xeons, but the effect is not as pronounced. Historically CDF code has been friendly to processors with larger cache. In this case the cache of the Opteron and Xeon are the same, and the memory bus is not an issue. The slope of speed improvement with frequency is larger as well.

Both CDF and D0 have said that they do not intend to recompile their code for use in 64-bit Linux environments, although both codes did at one time run in 64-bit SGI environments. The conclusion of this report is that even if legacy executables and a 32-bit OS are used, the 64-bit hardware results in improved performance. The Opterons show marginally better performance in this mode than the Xeons. Thus Opteron is a viable choice for Run II.

The Fermi Cycle benchmarks do not allow for the effects of hyperthreading. (Nor do the CMS benchmarks.) All the Xeon benchmarking data was run with hyperthreading disabled. Opterons do not have hyperthreading. We have done hyperthreading studies in the past with the same D0 executable and data file that were used in these benchmarks. Those studies showed that nodes just running two processes would finish them faster (about 10% faster) if hyperthreading was off. Total throughput was maximized (30% better than two processes in non-hyperthreaded mode) , however, if hyperthreading was on and four processes were running simultaneously. This is the mode used by the CDF CAF. The farms don't use hyperthreading because we want to make each individual process finish faster. For applications in which hyperthreading is going to be used, the Fermi Cycles per processor should be multiplied by 2.6 instead of by 2 for Xeon processors.

A by-product of this study is that the Fermi Cycle that we have used for Farms performance has been renormalized, and several of our existing machines now have better performance relative to the Pentium III 1 GHz than they used to have.

8.1.5 SDSS

SDSS found that their photo software ran only 6% slower on the Opteron 246HE with 4 GB of RAM than on the Opteron 250 with 2GB of RAM. Since space and power are both at a premium at Apache Point, the low voltage Opteron hardware is an ideal fit for them. The frame subtraction software ran 20% faster on the Opteron 250. Both Opteron machines they tested were significantly faster in processing their data and meet the test of being able to process one night's worth of data before it gets dark again the next night. On their existing 2GHz Xeon machine it takes 30.9 hours to complete the whole cycle of photo and frame subtraction. The Opteron 250 can complete the task in 13.4 hours.

8.1.6 Benchmark Summary

In all we found that the Opteron 246HE's performance was equivalent to its high voltage counterpart. Several sections of this study indicate there can be a 50% gain by using the 64-bit features of native compiled code, and using an optimizing compiler that can exploit the full instruction set.

The SPEC benchmarks of both Intel Xeon and AMD Opteron hardware are highly dependent on what compilers are used to compile them. It is difficult to know what numbers to use for comparison, because often several different SPEC results are published for the same chip and motherboard combination, while for other vendors' hardware there are no results at all. These results are usually run under Windows, using proprietary compilers and heap managers, and use heroic optimizations.

For example, the Xeon 3.0GHz 800FSB "Nocona" is a higher spec CINT2000 rating than the Xeon 3.06GHz 533FSB chip, but on legacy CDF code the 3.06GHz chip actually is faster by about the ratio of the two clock speeds.

8.2 Vendors and Hardware

Of the eight Opteron machines we received, six are acceptable for Fermilab. Passing are Penguin, Rackable, ASA, IBM, Koi, and Sun. Disqualified are CSI, due to repeated and fatal disk errors that they were unable to address, and HP because their initial unit was not to our specification in several respects and they were not able, and in some cases unwilling, to make the changes.

In addition, we find that the Dell Poweredge SC1425 is an acceptable dual Xeon server. It was necessary to evaluate it because Dell has rotated on to our approved list for Intel-based computer vendors and this unit is the unit they would use if they were to bid.

We are making arrangements with the vendors to purchase the seven different machines that we are keeping and send back the two that we are not keeping.

The roster of vendors for compute server buys now stands as follows: Intel: Koi, ASA, Rackable, CSI, Penguin, Dell. AMD: Koi, ASA, Rackable, Penguin, IBM, Sun. Of our vendor list that once stood at eighteen technically qualified vendors these are all that remain. We will need to reopen the vendor list in the near future to add more vendors to the roster.

Of the above vendors, we realize that either CD management or Business services may wish to disqualify ASA due to adverse experience in procurements not related to this evaluation, which was begun before the ASA problems became apparent. But no such decision has been made at this time by either party.

The Rackable unit as delivered required a choice between CD ROM drive or a second hard drive. The Sun unit has SCSI-based disk, it is possible to get SCSI disks big enough to fill our requirements but it could be quite expensive. Some of these units, the Sunfire v20z, have already been purchased at Fermilab and are being used as small database servers. Sun will be introducing a new model soon that is based on SATA drives and a Sun-produced motherboard. Both Rackable and Sun have requested to give presentations to CD staff to discuss plans for the upcoming procurements and how their hardware can best meet our needs. Koi has also alerted us to the fact that a SuperMicro Opteron board will soon be available and will get us a demo unit.

We suggest that we have a meeting at Fermilab including all of the above vendors to make presentations to them on what our needs are, show them our new facilities, and let them make presentations to us if they so desire. This should happen before the next RFP's go out if possible.

Three units in this evaluation, IBM, Dell, and ASA, had SATA disk drives (serial ATA), including drives from Seagate, Maxtor, and Western Digital. We did not experience any problems with any of these units and they all ran bonnie tests well and with good throughput. It should be noted, however, that the error checking and testing utilities for SATA drives are still rudimentary, with many of the functions of the smartmontools not being available for the kernel drivers we are currently using to drive them.

8.3 Budget

It is difficult to predict what the cost of worker nodes will be, based on the price of a single evaluation unit. Nevertheless we can make some estimates. There are four possible Opteron processors that are candidates, Opteron 246HE, 246, 248, and 250. There have been recent price reductions on all of these chips, they have each dropped by more than $150 over the last two months, since the Opteron 252 was just introduced. Based on current CPU prices, the Opteron 250 would now sell for about $2800. Changes in case configuration and removal of the add-on IPMI card could reduce this by $200 more. Changing to Opteron 248 would drop the price by $200/CPU, to $2200, to Opteron 246 would drop the price by $300/CPU, to $2000. Racking infrastructure, cabling, and power strips costs $7500 a rack nowadays. Thus a rack of conventional Opteron 246 could be ours for $87,500, 248's, for $95,500, 250's for $111,500. Corresponding infrastructure fractions of GCC, using 1.95,2.2,and 2.45A respectively for current draws, would be 1.3%, 1.46%, and 1.63%, $39000, $44000, and $49000 respectively.

The budget numbers above are based on patterns of where previous bids have come in. If the prices of the evaluation units and bid history of these vendors are any guide, the bids for the conventional opterons will span a range with the above numbers being the low bid and the median value being about 20-25% above that level. Thus if we expect to divide orders among more than one vendor, the appropriate contingencies should be added to account for the higher prices.

The Opteron 246HE unit is currently the only one available on the market that we knowof. The cost of the unit is $3850 when equipped with 4GB of RAM. This price for a single evaluation node is inflated, and the price of the Opteron 246HE, like the other chips has dropped significantly since that time. It is reasonable to assume that it can

be had for a 20% price premium per node over the conventional node, or approximately $2400/node, $103500/rack. Associated infrastructure costs would be 1%, or $32000.

For comparison, the Koi 3.0GHz/800FSB nodes that were recently deployed in GCC cost us $1840/node, $81,100 a rack, and use up 1.8% of GCC south room cooling infrastructure, $53750. Equivalent nodes could be had in today's market for less than $1700/node. We would expect that comparably-equipped 3.4GHz nodes could be had for $2100, and 3.6GHz nodes could be had for $2600/node.

Using Fermi Cycles as a basis for price-performance comparisons, we can make the following table, recognizing that price is an estimate at best, but the same pricing model is built into all these numbers.

Processor	Fermi Cycles	Projected Cost/Rack	FC/ dollar	FC/ Ampere	GCC Infra Cost/Rack	FC/dollar Corrected
Opteron 246	2622	87500	2.40	2689	39000	1.66
Opteron 246HE	2622	103500	2.03	3277	32000	1.55
Opteron 248	2915	95500	2.44	2650	44000	1.67
Opteron 250	3085	111500	2.21	2519	49000	1.54
Xeon 3.6(Dell)	2620	141460	1.48	1905	55000	1.07
Xeon 3.6(Koi)	2620	111500	1.88	1690	62000	1.21
Xeon 3.4	2480	91500	2.17	1710	58000	1.33
Xeon 3.0	2250	75500	2.38	1667	54000	1.39

Table 12: *Performance/price.*

From the table, it appears that the best total price/performance, correcting for the infrastructure, is a tie between the conventional Opteron 246 and 248. Since prices are likely to continue to drop on the 248 in the time before the bid, the 248 is preferred for pure price/performance and also for infrastructure-adjusted price performance. For the Opteron 246HE to be competitive it would have to be sold at about the same price as the Opteron 248, $2200/node.

For applications that require Intel processors, the price-performance sweet spot is now at 3.2 GHz. This is the frequency that LQCD intends to buy in their next procurement. Applications that make heavy use of hyperthreading (CDF CAF) should consider staying with Intel hardware as well.

8.4 Procurement Recommendations

There are several upcoming acquisitions: for General Purpose Farms, CMS, Run II, and SDSS at Apache Point. For these acquisitions we believe a price performance bid is not the way to go. We should rather pick one configuration, specify a single disk configuration and a single CPU speed. Based on the arguments above, we recommend that this be the Opteron 248. Apache Point may consider the Opteron 246HE, particularly since their benchmarking shows that they can run just as well on a slower machine if it has bigger memory.

Given the newness of the technology we should arrange to make single-rack orders to a number of the vendors so that we can get some experience with the technology before the large Run II procurements at the end of the fiscal year. Also, as mentioned above, we strongly recommend bringing all six vendors to Fermilab before the bids are sent out so that they are well aware of our facility and racking requirements.

8.5 Future developments

Dual core CPU's are announced for 2005. They promise to nearly double the computing power while not increasing the demands for space, cooling and electricity significantly. It will be very interesting to test these CPU's once they become available. In general with the applications studied in this report we observe no drop in efficiency when running 2 processes simultaneously on the 2 CPU's. This is a good indication that multi-core chips will just scale with the number of processors for our applications. We expect dual core cpu's to be available for both the Intel and AMD 64-bit chips by the end of calendar year 2005.

As mentioned above, new motherboards and technology continue to be developed in the Opteron space as well as the Xeon space. The recently released Pentium 4 600 series chips show a decrease in power, giving hope that this innovation will eventually spread to the Xeon space as well.

References

[1] **Proceedings of CHEP 2004 Interlaken, Switzerland**, P. Wegner, S.Wiesand, *"64-Bit Opteron systems in High Energy and Astroparticle Physics"*.

[2] http://cmsdoc.cern.ch/cmsoo/cmsoo.html.

[3] *DAR was developed by Natalia Ratnikova and stands for:* **D***istribution* **A***fter* **R***elease*, http://home.fnal.gov/∼natasha/DAR/dar.html.

[4] http://root.cern.ch.

[5] T. Sjöstrand, P. Edén, C. Friberg, L. Lönnblad, G. Miu, S. Mrenna and E. Norrbin, Computer Phys. Commun. 135 (2001) 238 (LU TP 00-30, hep-ph/0010017).

[6] *"What's So Great for Developers About the AMD64?"* David Kaplowitz June 23, 2003 http://www.devx.com/amd/Article/16018.

[7] *"64-bit Computing with Intel EM64T and AMD AMD64"* http://www.redbooks.ibm.com/abstracts/tips0475.html.